

AMENDMENTS TO THE CLAIMS

The listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently Amended) A computer system for selecting and organizing individual test cases from a binary program module of test cases for use in testing a computer program to ensure that the program processes as intended, the system comprising:

a binary program module ~~containing~~ storing a plurality of individually accessible test cases, each comprising a set of instructions for testing a feature of the computer program through a language and format independent interface, at least some of the individually accessible test cases differing from one another in language and format;

a harness comprising a set of instructions that executes a test case hierarchy on the computer program using the corresponding language and format independent interface of each individually accessible test case in the test case hierarchy;

B 1
a connector comprising a set of instructions that scans the plurality of test cases and extracts those test cases to be used to test the computer program to ~~insure~~ ensure that it processes as intended, the connector creating a hierarchy of test cases from those that are selected and extracted, and selectively integrating an interface between the test case hierarchy and the harness regardless of the language or format in which the test cases were written; and

a processor for executing the one or more test cases, the harness and the connector.

2. (Previously Presented) A computer system as recited in claim 1, wherein the set of instructions of the harness and the set of instructions of the connector utilize an architecture that defines a means for accessing a resource over a network.

3. (Previously Presented) A computer system as recited in claim 2, wherein the architecture is COM technology.

4. (Currently Amended) In a computer system that includes a processor, a computer program to be tested, a binary program module ~~of storing~~ a plurality of individually accessible test cases for use in testing the computer program, a harness for executing individual test cases on the computer program, and a connector for interfacing between the binary program module and the harness, a method for selecting and extracting test cases from the binary program module for use in testing the computer program to determine whether the computer program processes as intended, the method comprising the steps for:

the connector scanning the binary program module storing the plurality of individually accessible test cases, at least some of which differ from one another in language and format, for one or more test cases of interest, each test case having a language and format independent interface for executing the test case on the computer program regardless of the language or format used to develop the test case;

the connector extracting the one or more test cases of interest from the binary program module;

the connector organizing one or more test cases into a test case hierarchy;

the connector interfacing a harness with the one or more test cases of interest, wherein the interfacing allows the harness to recognize and execute the one or more test cases of interest regardless of the language or format in which the one or more test cases of interest were developed; and

the harness traversing the test case hierarchy and executing each of the one or more test cases of interest to test the computer program.

Claims 5 – 6 (cancelled).

7. (Original) A method as recited in claim 4, wherein the method further includes the step for determining whether one or more of the test cases are identified as being deselected, wherein a deselected test case is not executed on the computer program.

8. (Original) A method as recited in claim 7, wherein one or more test cases comprise a test suite in the hierarchy.

9. (Original) A method as recited in claim 8, wherein one or more test suites comprise a test module in the hierarchy.

10. (Original) A method as recited in claim 9, wherein upon the user selecting a test suite, the one or more test cases that comprise the test suite, excluding any test cases determined to be deselected are selected.

11. (Original) A method as recited in claim 10, wherein upon the user selecting a test module, the one or more test suites that comprise the test module, excluding any test cases determined to be deselected are selected.

12. (Previously Presented) A method as recited in claim 4, wherein the step for traversing further includes executing the one or more test cases on a thread pool comprising one or more threads.

13. (Original) A method as recited in claim 12, wherein the step for traversing further includes copying a selected test case across all of the one or more threads, and wherein the selected test case is executed across all of the one or more threads.

14. (Previously Presented) A method as recited in claim 12, wherein the step for traversing further includes executing a selected test case on one of the threads.

15. (Currently Amended) A computer program product for implementing within a computer system a method for testing a computer program to determine whether the computer program processes as intended, the computer program product comprising:

computer readable medium for providing computer program code means utilized to implement the method, wherein the computer program code means is comprised of executable code for implementing the steps for:

scanning a binary program module that stores a plurality of individually accessible test cases, at least some of which differ from one another in language and format, for one or more test cases of interest to be used in testing the computer program, each test case having a language and format independent interface for executing the test case on the computer program regardless of the language or format used to develop the test case;

extracting the one or more test cases of interest from the binary program module;

organizing the one or more test cases of interest into a test case hierarchy;

interfacing with the one or more test cases of interest to recognize and execute the one or more test cases of interest regardless of the language or format in which the one or more test cases of interest were developed; and

traversing the test case hierarchy to execute each of the one or more test cases of interest on the computer program, regardless of the language or format in which the one or more test cases of interest were written.

16. (Original) A computer program product as recited in claim 15, wherein the step for traversing is performed by the harness.

17. (Original) A computer program product as recited in claim 15, wherein the step for interfacing is performed by one or more connectors.

18. (Original) A computer program product as recited in claim 15, wherein the step for organizing the test cases into a hierarchy includes grouping one or more of the test cases into a test suite.

19. (Original) A computer program product as recited in claim 18, wherein the step for organizing the test cases into a hierarchy includes grouping one or more test suites into a test module.

20. (Original) A computer program product as recited in claim 15, wherein the one or more test cases executed on the computer program are selected by a user.

21. (Previously Presented) A computer program product as recited in claim 20, wherein the step for traversing further includes executing the one or more test cases on a thread pool comprising one or more threads.

22. (Original) A computer program product as recited in claim 21, wherein the step for traversing further includes copying a selected test case across all of the one or more threads, and wherein the selected test case is executed across all of the one or more threads.

23. (Previously Presented) A computer program product as recited in claim 21, wherein the step for traversing further includes executing a selected test case on one of the threads.

24. (Currently Amended) In a computer system that includes a computer program to be tested, a program module of test cases written in any format or language for testing the computer program, a harness for executing one test cases on the computer program, and one or more connectors for interfacing test cases with the harness, a method for testing the computer program to determine whether the computer program processes as intended, the method comprising the acts of:

identifying one or more test cases from a binary program module that stores a plurality of individually accessible test cases, at least some of which differ from one another in language and format, each test case implementing a language and format independent interface for executing the test case on a computer program regardless of the language or format used to develop the test case;

translating the identified one or more test cases into a test case hierarchy; ~~and~~

interfacing the test case hierarchy in order to recognize and execute the one or more test cases regardless of the language or format in which the one or more test cases were written; and

executing each of the one or more test cases in the test case hierarchy to test the computer program regardless of the language or format in which the one or more test cases were written.

25. (Original) A method as recited in claim 24, wherein the act of executing is performed on a thread pool comprising one or more threads.

26. (Previously Presented) A method as recited in claim 25, wherein the act of executing is further performed by executing a selected test case on one of the threads.

27. (Previously Presented) A method as recited in claim 25, wherein the act of executing is further performed by copying each of the one or more test cases across all of the one or more threads, and wherein each test case is executed across all of the one or more threads.